

CLAIMS

What is claimed is:

- Sub
a1
- 10006477.1.13001
- 1 1. A method for buffering data in a computer graphics pipeline, comprising:
 - 2 (a) producing graphics floating point data in a graphics pipeline;
 - 3 (b) operating on the graphics floating point data in the graphics pipeline; and
 - 4 (c) storing the graphics floating point data to a buffer in the graphics pipeline;
 - 5 (d) wherein the graphics floating point data is read and stored in an unclamped
 - 6 format for increasing a parameter selected from the group consisting of a
 - 7 precision and a range of the graphics floating point data.
 - 1 2. The method as recited in claim 1, wherein the graphics floating point data
 - 2 includes fragment data.
 - 1 3. The method as recited in claim 2, wherein the fragment data is received from a
 - 2 rasterizer.
 - 1 4. The method as recited in claim 2, wherein the fragment data includes color data.
 - 1 5. The method as recited in claim 2, wherein the fragment data includes depth data.
 - 1 6. The method as recited in claim 1, wherein the graphics floating point data is
 - 2 only constrained by an underlying data type.
 - 1 7. The method as recited in claim 1, wherein the buffer serves as a texture map.

- 1 8. A computer program product for buffering data in a computer graphics pipeline,
2 comprising:
3 (a) computer code for producing graphics floating point data in a graphics pipeline;
4 (b) computer code for operating on the graphics floating point data in the graphics
5 pipeline; and
6 (c) computer code for storing the graphics floating point data to a buffer in the
7 graphics pipeline;
8 (d) wherein the graphics floating point data is read and stored in an unclamped
9 format for increasing a parameter selected from the group consisting of a
10 precision and a range of the graphics floating point data.
- 1 9. The computer program product as recited in claim 8, wherein the graphics
2 floating point data includes fragment data.
- 1 10. The computer program product as recited in claim 9, wherein the fragment data
2 is received from a rasterizer.
- 1 11. The computer program product as recited in claim 9, wherein the fragment data
2 includes color data.
- 1 12. The computer program product as recited in claim 9, wherein the fragment data
2 includes depth data.
- 1 13. The computer program product as recited in claim 8, wherein the graphics
2 floating point data is only constrained by an underlying data type.
- 1 14. The computer program product as recited in claim 8, wherein the buffer serves
2 as a texture map.

100647-113001
FOUO

- 1 15. A system for buffering data in a computer graphics pipeline, comprising:
2 (a) logic for producing graphics floating point data in a graphics pipeline;
3 (b) logic for operating on the graphics floating point data in the graphics pipeline;
4 and
5 (c) logic for storing the graphics floating point data to a buffer in the graphics
6 pipeline;
7 (d) wherein the graphics floating point data is read and stored in an unclamped
8 format for increasing a parameter selected from the group consisting of a
9 precision and a range of the graphics floating point data.

- 1 16. A buffering apparatus in a computer graphics pipeline, comprising:
2 (a) a buffer capable of storing graphics floating point data in a graphics pipeline;
3 (b) wherein the graphics floating point data is stored in an unclamped format for
4 increasing a parameter selected from the group consisting of a precision and a
5 range of the graphics floating point data.

- 1 17. A system for buffering data in a computer graphics pipeline, comprising:
2 (a) means for producing graphics floating point data in a graphics pipeline;
3 (b) means for operating on the graphics floating point data in the graphics pipeline;
4 and
5 (c) means for storing the graphics floating point data to a buffer in the graphics
6 pipeline;
7 (d) wherein the graphics floating point data is read and stored in an unclamped
8 format for increasing a parameter selected from the group consisting of a
9 precision and a range of the graphics floating point data.

- 1 18. A method for buffering data in a computer graphics pipeline, comprising:
2 (a) producing graphics floating point data in a graphics pipeline;
3 (b) operating on the graphics floating point data in the graphics pipeline; and

- 4 (c) storing the graphics floating point data to a buffer in the graphics pipeline;
5 (d) wherein the buffer serves as a texture map.
- 1 19. A buffering apparatus in a computer graphics pipeline, comprising:
2 (a) a buffer capable of storing graphics floating point data in a graphics pipeline;
3 (b) wherein the buffer serves as a texture map.
- 1 20. A method for buffering data during multi-pass rendering in a computer graphics
2 pipeline, comprising:
3 (a) operating on graphics floating point data during a rendering pass in a graphics
4 pipeline;
5 (b) reading the graphics floating point data from a buffer during the rendering pass
6 in the graphics pipeline;
7 (c) storing the graphics floating point data to the buffer during the rendering pass in
8 the graphics pipeline; and
9 (d) repeating (a) – (c) during additional rendering passes.
- 1 21. The method as recited in claim 20, wherein the operating includes deferred
2 shading.
- 1 22. A method for buffering data in a computer graphics pipeline, comprising:
2 (a) producing graphics floating point data in a graphics pipeline;
3 (b) packing the graphics floating point data in the graphics pipeline; and
4 (c) storing the graphics floating point data to a buffer in the graphics pipeline.
- 1 23. A method for buffering data in a computer graphics pipeline, comprising:
2 (a) producing graphics floating point data in a graphics pipeline;
3 (b) unpacking the graphics floating point data in the graphics pipeline; and
4 (c) operating on the unpacked graphics floating point data in the graphics pipeline.

- 1 24. A method for buffering data in a computer graphics pipeline, comprising:
- 2 (a) operating on graphics floating point data in a graphics pipeline;
- 3 (b) producing the graphics floating point data in the graphics pipeline; and
- 4 (c) storing the graphics floating point data to a buffer in the graphics pipeline;
- 5 (d) wherein the graphics floating point data is read and stored in an unclamped
- 6 format for increasing a parameter selected from the group consisting of a
- 7 precision and a range of the graphics floating point data.

10006477-113001